15th Triennial World Congress, Barcelona, Spain

MODEL PREDICTIVE CONTROL OF A CHEMICAL PROCESS BASED ON AN ADAPTIVE NEURAL NETWORK

D. L. Yu, D. W. Yu, J. B. Gomm and D. Williams

Control Systems Research Group, School of Engineering
Liverpool John Moores University, Byrom Street, Liverpool L3 3AF
E-mail: d.yu@livjm.ac.uk

Abstract. An adaptive neural network-based predictive strategy is applied to a pilot multivariable chemical reactor. The first stage of the project, simulation study, has been investigated and is presented in this paper, together with the description of the adaptive network. A pseudo-linear radial basis function (PLRBF) network is developed to model the real process and its weights are on-line updated using a recursive orthogonal least squares (ROLS) algorithm. The effectiveness of the adaptive control in improving the closed-loop performance has been demonstrated for process time-varying dynamics and model-process mismatch.

Keywords. Adaptive neural networks, multivariable systems, model predictive control.

1. INTRODUCTION

Applications of neural networks in chemical process modelling and model predictive control (MPC) have been investigated for single-input, single-output systems (Lightbody and Irwin, 1997; Doherty *et al*., 1997). A neural network modelling and MPC control technique for multivariable process was investigated in our previous research and described in (Yu et al. 1999), where three variables including temperature, pH and dissolved oxygen were controlled for set-point tracking in real time. In the on-line control practice, however, it was realised that the dissolved oxygen in the reaction had time-varying dynamics which significantly degraded the on-line performance when a fixed neural network model was used. This is the motivation behind this research to develop an adaptive neural model to cope with the time-varying dynamics and also the model-process mismatch.

Different adaptive neural networks have been developed in recent years. For example, Lu et al. (1997), Karayiannis and Mi (1997), Luo and Billings (1998) proposed different adaptation algorithms for RBF network structure to recursively train the network model in off-line mode or to model a time varying system in on-line mode. Liu et al. (1999) developed an adaptive RBF network and Yang and Linkens (1994) developed an adaptive multi-layer perceptron (MLP) network as the adaptive controllers for non-linear system control. Pereira et al. (2000) applied adaptive RBF network model in the internal model control strategy to control an experimental process, and compared the performance with that achieved using a linear pole-placement controller.

A weight and centre adapted RBF network has been developed by the authors (Yu and Gomm, 2001) based on the centre pruning algorithm (Gomm and Yu, 2000) and ROLS training algorithm (Yu et al., 1997). In this paper a pseudo-linear RBF network is on-line trained using ROLS as the process model and is used in MPC of a laboratory-scaled chemical reactor. The reactor exhibits characteristics typical of many industrial processes, due to its non-linearity, coupling effects among the controlled variables and a long time-delay in heat exchange. The work in the first stage, simulation study, is described in the paper. It will be followed by the work in the second stage, real time application to the reactor and the application of the both weight and centre adapted RBF network model.

The structure of the pseudo-linear RBF is described in the paper. Then, a brief review of the on-line operation of the ROLS training algorithm follows. In the simulation part, an MLP network is trained using the Levenberg-Marquardt algorithm and used as the simulated process, due to the lack of the process mathematical model. Then a change in process dynamics is simulated during on-line control to demonstrate the effectiveness of the adaptive model in coping with time varying dynamics. This performance is compared with that by the same MPC with the same but fixed RBF model.

2. PROCESS DESCRIPTION

The reactor used in this research is a pilot system established in the laboratory to generally represent the dynamic behaviour of real chemical processes in industry. The schematic of the chemical reactor is shown in Fig.1. It consists of a continuously stirred tank (15 litres) to which the chemical solutions, NH_4OH, CH_3COOH and $N_{a2}SO_3$, and air are added. The liquid level in the tank is maintained at a pre-specified constant level by an outflow pump system. The concentrations and flow rates of solutions, CH_3COOH and $N_{a2}SO_3$, are constant except for some manual changes to mimic process disturbances. The concentration of NH_4OH is constant but the flow rate is adjustable by a servo-pump to regulate the pH value in the tank. The air-flow rate is also adjustable by a mass- flow meter connected to a compressing air network to regulate the percentage of the dissolved oxygen (pO_2) in the liquid in the tank. The tank is also equipped with an electric heating system to adjust the liquid temperature. The liquid in the tank is stirred continuously to make sure the pH, the dissolved oxygen and the temperature are consistent throughout the tank. All three variables are measured and displayed. A personal computer with analogue I/O is connected to the process to sample the measurements and issue the control outputs.

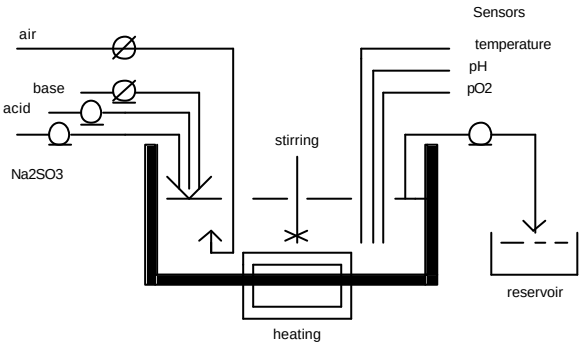


Fig.1 The chemical reactor process

With the three inputs, heating power, flow rate of ammonium hydroxide and flow rate of air, and the three outputs, liquid temperature, pH and percentage of dissolved oxygen, the process constitutes a MIMO, non-linear dynamic system. It has been shown in the experiments that the coupling between variables is very significant. The rate of absorption of oxygen into the liquid and the reaction of the sodium sulphite, for example, significantly depend on the liquid temperature. The process also suffers from many external disturbances, apart from those introduced manually, such as changes in the room temperature, perturbations in the concentrations of the inflow chemical solutions and air pressure in the compressing air network, different concentrations of H^+ and OH^- ions in the liquid at different times. In addition, the response times for the three variables are significantly different. The rise time for the temperature is very long due to the available heating power whereas the dissolved oxygen is quite short. All these effects cause the process to be non-linear in both dynamic and static behaviour, time varying and uncertain in parameters, multivariable with significant coupling, complex without a known mathematical model, suffering from unpredictable large disturbances.

Process inputs and outputs are chosen as

$$u = \begin{bmatrix} Q \\ f_b \\ f_a \end{bmatrix}, \qquad y = \begin{bmatrix} T \\ pH \\ pO_2 \end{bmatrix} \tag{1}$$

where Q, f_b and f_a denote the heating power, the flow rate of the base and the flow rate of air respectively.

One possibility for choosing a sample time is based on examining the hold-up time (volume/total flow rate) of the process, which is approximately 10 minutes for this process. A suitable sample time could then be selected as, say, one minute if only the dynamics of the fluid flow rates are considered. However, since the dissolved oxygen responds to changes in the air flow rate quickly, especially at high temperature, therefore, the rise times for different variables should also be considered. From process step responses, approximate rise times to reach steady state are recognised as 45 minutes for temperature, 25 minutes for pH, 5 minutes for pO_2 when temperature is 30^0C and 2 minutes for pO_2 at 50^0C. Thus, a suitable sample interval for all variables was selected to be 10 seconds.

3. NEURAL NETWORK MODELS

The process is represented by the multivariable NARX model of the following form,

$$\begin{aligned} \hat{y}(k) = & f(y(k-1),\cdots,y(k-n_y), \\ & u(k-1-d),\cdots,u(k-n_u-d)) + e(k) \end{aligned} \tag{2}$$

where

$$y(k) = \begin{bmatrix} y_1(k) \\ \vdots \\ y_p(k) \end{bmatrix}, \; u(k) = \begin{bmatrix} u_1(k) \\ \vdots \\ u_m(k) \end{bmatrix}, \; e(k) = \begin{bmatrix} e_1(k) \\ \vdots \\ e_p(k) \end{bmatrix}$$

are the process output, input and noise respectively, p and m are the number of outputs and inputs respectively, n_y and n_u are the maximum lags in the outputs and inputs respectively, d is the maximum time delay in the inputs; and $f(*)$ is a vector-valued, non-linear function. When neural networks are used to model the process, the measurements of the process at different sample times can be taken as the

input of the network, while the network implements the non-linear transformation, $f(*)$ in (2).

A PLRBF network is proposed in this work by augmenting the nominal RBF with the network input directly used as part of the regression signals. The PLRBF structure and operation are presented by the following equations.

$$f = \begin{bmatrix} z^T & x^T \end{bmatrix}^T \quad (3)$$

$$z_i = \exp(-\frac{\|x - c_i\|^2}{s^2}), i = 1, \cdots, n_h - n \quad (4)$$

$$\hat{y} = W^T f \quad (5)$$

where $x, c_i \in \Re^n$ with $n = pn_y + mn_u$ are the input vector to the network and the centre vector in the hidden layer, $f \in \Re^{n_h}$ is the hidden layer output vector, $W \in \Re^{p \times n_h}$ is the output layer weight matrix and $\hat{y} \in \Re^p$ is the network output, or the prediction of the process output when the network is used as the process model. The network input vector is chosen as $x(k) = \left[y^T(k-1), \cdots, y^T(k-n_y), u^T(n-1-d), \cdots, u^T(k-1-n_u) \right]^T$ according to (2). It is noted that such structured PLRBF is a combination of a linear part and a non-linear part in one model. This reduces the task of the hidden layer nodes to modelling only additional non-linear effects to a linear model throughout the whole operating region. Simulation studies have shown that much less hidden layer nodes are needed than a standard RBF to model a process to the same accuracy.

The weight matrix of the PLRBF is trained initially and then on-line updated using the ROLS algorithm. For the N input-output training data, it is formed according to (5)

$$Y = \hat{Y} + E = \Phi W + E \quad (6)$$

where $Y \in \Re^{N \times p}$ is the desired output matrix, $\hat{Y} \in \Re^{N \times p}$ is the neural network output matrix, $\Phi \in \Re^{N \times n_h}$ is the hidden layer output matrix, $E \in \Re^{N \times p}$ is the error matrix. $W(k)$ can be solved from

$$R(k)W(k) = \hat{Y}(k) \quad (7)$$

where R(k) is from the QR decomposition of Φ. Calculation of W can be achieved on-line using the following transformation (Bobrow and Murray, 1993),

$$\begin{bmatrix} lR(k-1) & l\hat{Y}(k-1) \\ f^T(k) & y^T(k) \end{bmatrix} \rightarrow \begin{bmatrix} R(k) & \hat{Y}(k) \\ 0 & h^T(k) \end{bmatrix} \quad (8)$$

The procedure of the ROLS algorithm is therefore the following: at stage t, calculate $R(t)$ and $\hat{Y}(t)$ according to (8), then solve $W(k)$ in (11). Initial values for $R(k)$ and $\hat{Y}(k)$ can be assigned as $R(0) = aI$ and $\hat{Y}(0) = 0$, where a is a small positive number.

In this study, a PLRBF is trained as the process model and a MLP is trained as the process simulation for on-line control evaluation. The process time-delay and input-output orders were selected in the previous work using a linearized selection method (Yu et al., 2000). The network input vector is chosen accordingly as,

$$x(k) = [T(k-1), T(k-2), pH(k-1), pH(k-2), pO_2(k-1), pO_2(k-2), Q(k-1-d), f_b(k-1), f_a(k-1)]^T \quad (9)$$

where d = 22 is the time-delay from the heating power to the liquid temperature.

A training data set with 1800 samples is collected from the process. It is found that the data displays a time variant feature with the first 200 samples of dissolved oxygen behaving much different from the rest of the samples. It is thus decided the PLRBF model is trained using the last 1600 samples, while the MLP network is trained using the whole data set to present the process dynamics. Worth mentioning is that all data is scaled to the region, 0~1 in training and prediction in on-line control simulation. The trained PLRBF model is used as a parallel model to predict the process output used in the training, and the prediction is displayed in Fig.2.

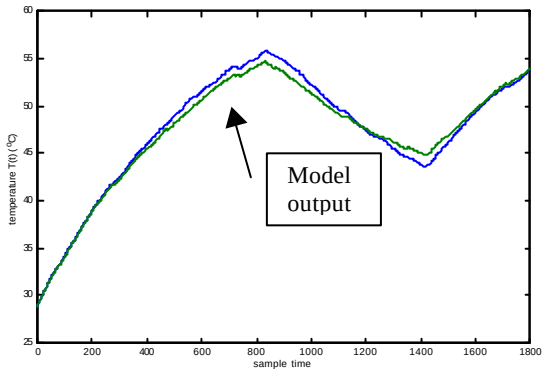


Fig.2a Process and parallel model output for temperature

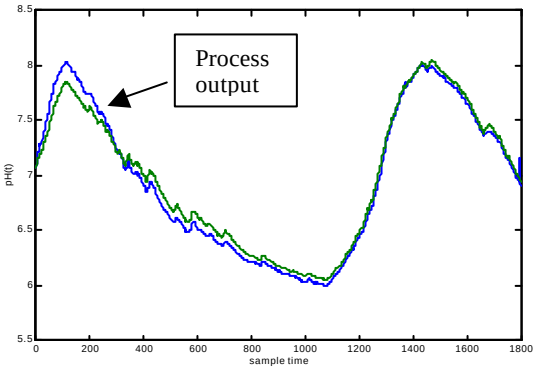


Fig.2b Process and parallel model output for pH

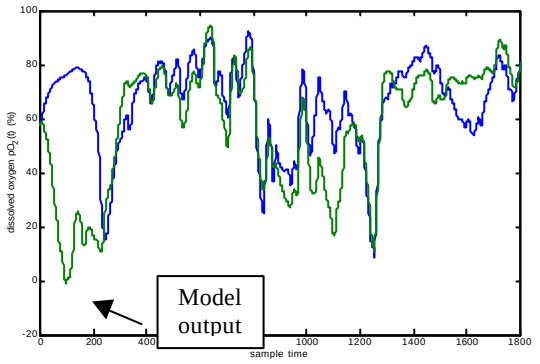


Fig.2c Process and parallel model output for dissolved oxygen

It can be observed in Figs 2a-2c that the prediction of the PLRBF model in the parallel format is quite accurate for temperature and pH, and is generally well for dissolved oxygen except for the first 200 samples. This is because the first 200 samples have a quite different dynamic behaviour due to the process time variant dynamics and are not used to train this model. The MLP model trained for use as a simulated process is also accurate but is not displayed here due to the limited space. As the MLP model is trained by a training set including the first 200 samples of dissolved oxygen, the two models are actually have a mismatch that will be used to evaluate the developed adaptive control.

4. MODEL PREDICTIVE CONTROL

Model predictive control using a neural network model for single-input, single-output systems has been studied by a few researchers and is outlined in by Hunt *et al*. (1992). For multivariable systems the neural network MPC strategy used to control this reactor was described in the previous research (Yu et al., 1999) using three fixed MLP models. The same strategy is used here with an adaptive model and is shown in Fig.3. In this approach, a neural network model is used to predict the future process response over the specified horizon. The predictions are passed to a numerical optimisation routine which attempts to minimise a specified cost function (10) in the calculation of a suitable control signal at each sample instant.

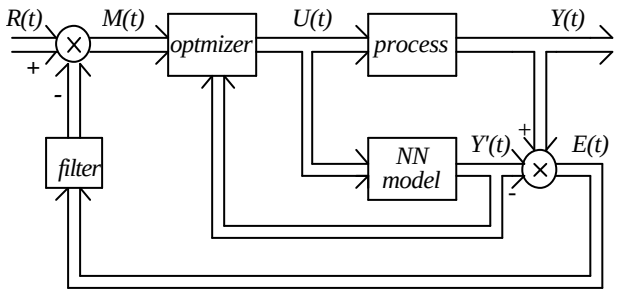


Fig.3 Multivariable MPC control strategy

The objective function used in the optimisation is

$$J=\sum_{i=1}^{p}(M_i-\hat{Y}_i)^T W_y^{(i)}(M_i-\hat{Y}_i)+\sum_{j=1}^{m}x(j)\Delta U_j^T W_u^{(j)}\Delta U_j \quad (10)$$

where

$$M_i^T=\left[m_i(t+N_1(i)),\cdots,m_i(t+N_2(i))\right]$$

$$\hat{Y}_i^T=\left[\hat{y}_i(t+N_1(i)),\cdots,\hat{y}_i(t+N_2(i))\right]$$

$$\Delta U_j^T=\left[u_j(t)-u_j(t-1),\cdots,\right.$$
$$\left.u_j(t+N_u(j))-u_j(t+N_u(j)-1)\right]$$

N_1, N_2 are vectors specifying the prediction horizon, with the *i*th element specifying the parameter for the corresponding output, N_u is the control horizon vector, with the *j*th element specifying the parameter for the corresponding input. The second summation term in the cost function (10) is designed to smooth the control signal. $W^{(i)}_y$ and $W^{(j)}_u$ are the weighting matrices for the ith output tracking error and jth input increment, respectively. x is the control weighting vector, with the *j*th element applied to the *j*th input. M is the modified set-point to introduce a feedback in order to compensate the system steady-state error and against disturbance effects. The filtered model prediction error in Fig.3 is also used to compensate the model outputs in the MPC scheme to correct the future predictions. This is to modify the model outputs in (5) by adding the corresponding filtered error to the right-hand side of the equations.

Before on-line evaluation, the control system is simulated using the developed MLP model as the process. In this way, the system can be easily operated repeatedly to find a set of suitable control parameters, i.e. values for the vectors N_1, N_2, N_u, x. A number of simulations have been tried and suitable control parameters for set point tracking are chosen as, $N_1=[22 \quad 2 \quad 6]^T$, $N_2=N_1+[15 \quad 15 \quad 15]^T$, $N_u=[1 \quad 1 \quad 1]^T$, $x=[0.05 \quad 0.05 \quad 0.05]^T$, where the elements of N_1 and N_2 relate to the temperature (1), pH (2) and dissolved oxygen (3) respectively, and the elements of N_u and ξ relate to the heating power (1), flow-rate of base (2) and flow-rate of air (3) respectively. Here the value of $N_1(1)$ for the temperature must equal to or be greater than the time-delay $d=22$. $N_1(2)=2$ and $N_1(3)=6$ are chosen to consider the future output tracking and to achieve a smooth control. The choice of N_2 includes an equal number of future predictions of each output in the cost function which, with these settings of N_1 and N_2, is 15. If the elements of N_u are greater than 1, the number of variables to be optimised will not be minimum. Since the optimisation is multi-dimensional, this increases the difficulty and computational expense required to solve the optimisation. Hence, $N_u=1$ only is considered in this work. Only a small damping on the control inputs ($\xi=0.01$) was found necessary with these relatively long prediction horizon settings since the control

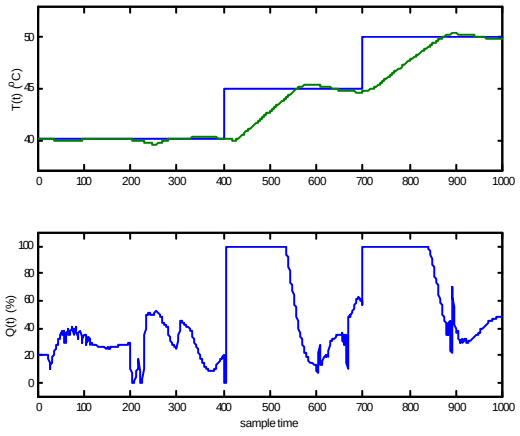
inputs were not observed to exhibit undue oscillations. Sequential quadratic programming is used to solve the multivariable optimisation problem of minimising J with respect to u(t), and to produce a solution constrained within the process input operating ranges (0~100% of the scaled data).

The on-line adaptation of the model is performed by conducting (8) using the Givens rotation to the augmented matrix, then solve weight matrix W(k) in (7) by back substitution. A forgetting factor $l = 0.98$ is found appropriate. To prevent the model losing useful past information in the period when the system dynamics does not change significantly, the following condition is applied to determine whether the weight is adapted at this sample instant.

$$\frac{1}{L}\sum_{i=1}^{L}\left\| y(k+L-i) - \hat{y}(k+L-i)\right\| > d \qquad (11)$$

where L is the length of the window in which the mean modelling error norm is observed, d is a pre-specified threshold which is given corresponding to the model accuracy. In this research, L=5 and d =0.005 are used.

In order to evaluate the performance of the adaptive model based MPC, a dynamics change of the process is simulated at sample time instant k=50 by changing some weights of the MLP model with 10%. This simulated model with the change is controlled twice, once by the adaptive model and once by the same model without adaptation, with all other control parameters being the same. The system response together with associated control variable for three variables are displayed in Fig.4. The response of the simulated process for the model without adaptation is displayed in Fig.5 for comparison. Due to the limited space only the dissolved oxygen is displayed.



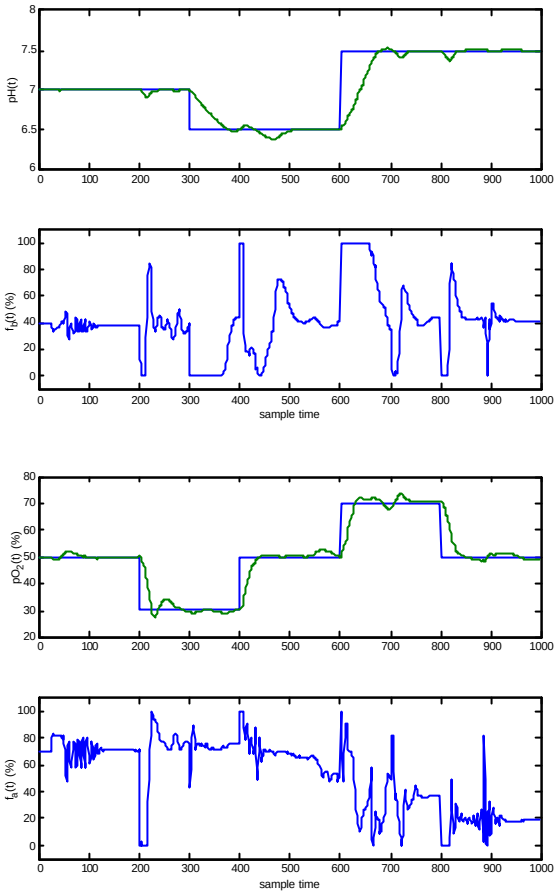


Fig.4 Simulated response and control with adaptive model

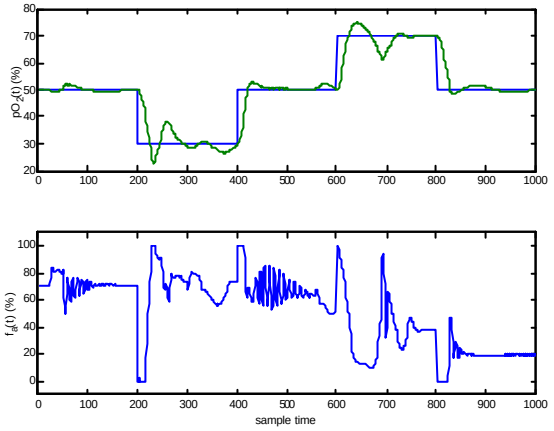


Fig.5 Simulated dissolved oxygen with fixed model

It is observed in Fig.4 that all three variables follow the set-point with quick response (with bounded control) and a very small overshoot even for the temperature which has a long time-delay. This is due to the multi-step ahead prediction considered in the objective function. There is no steady-state error to appear. Compared to the response of the dissolved oxygen with the adaptive model, the response produced with fixed model has a much worse performance. This demonstrates the effectiveness of the model adaptation to trace process time varying dynamics. It can also be seen that when one control

variable has an abrupt change to respond set-point change, the change is in fact a disturbance to the other outputs. Since the multivariable optimisation considers squared tracking errors of all variables, these disturbances are efficiently rejected by determining changes in the other control variables for compensation.

The performance of the three variables is also compared numerically with measurement of the mean-squared-error of set-point tracking and on-line model prediction and displayed in Table 1.

	MSE for pO_2		MSE for all outputs	
	Tracking	Modelling	Tracking	Modelling
Adaptive model	0.0020	1.1324 e-7	0.0095	2.6153 e-7
Fixed model	0.0022	7.4740 e-7	0.0101	2.3363 e-6

Table 1 Comparison of control performance

The MSEs in the table show that the on-line modelling error is greatly reduced by on-line training. The tracking performance is also improved in the adaptive model case.

5. CONCLUSIONS

Model predictive control based on an adaptive PLRBF model is applied to a laboratory-scaled three-input three-output chemical reactor. The PLRBF uses much less hidden layer nodes while predicts much more accurately than a standard RBF network when they are used to model multivariable real processes. On-line update of RBF weights using the ROLS and implemented by applying the Givens rotation is numerically stable and computing efficiently. Simulation results show that the adaptive model significantly reduces the instant modelling error for a time variant process or process-model mismatch, and consequently improves tracking error of the control system.

The work presented in this paper is the first stage of the project. The further work will focus on using a PLRBF model with its number of centres also adapted, and real-time evaluation on the process.

ACKNOWLEDGEMENTS

This work is supported by the UK Engineering & Physical Science Research Council under Grant Reference Number: GR/N 18697.

REFERENCES

Doherty, S. K., Gomm, J.B. and Williams, D., 1997, Experiment design considerations for non-linear system identification using neural networks. *Computers and Chemical Engineering*, Vol.21, No.3, pp 327-346.

Gomm, J.B. and Yu, D.L., 2000, Selecting radial basis function network centres with recursive orthogonal least squares training, *IEEE Trans. Neural Networks*, Vol.11, No.2, pp. 306-314.

Hunt K.J., Sbarbaro D., Zbikowski R. and Gawthrop P.J., 1992, Neural networks for control systems -- a survey. *Automatica*, Vol.28, No.6, pp 1083-1112.

Karayiannis, N.B. and Mi, G.W., Growing radial basis neural networks: merging supervised and unsupervised learning with network growth techniques, *IEEE Trans. Neural Networks*, Vol.8, No.6, pp. 1492-1506.

Lightbody, G. and Irwin, G.W., 1997. Non-linear control structures based on embedded neural system models. *IEEE Trans. Neural Networks*, Vol.8, No.3, pp 553-567.

Liu, G.P., Kadirkamanathan, V. and Billings, S.A., 1999, Variable neural networks for adaptive control of nonlinear systems, *IEEE Trans. Sys.Man.Cyber*. Pert.C, Vol.29, No.1, pp. 34-43.

Luo, W and Billings, S.A., 1998, Structure selective updating for nonlinear models and radial basis function networks, *Int.J.Adap.Cont.Sign.Proces*., Vol.12, pp.325-345.

Pereira, C., Henriques, J and Dourado, A., 2000, Adaptive RBFNN versus conventional self-tuning: comparison of two parametric model approaches for non-linear control, *Control Engn. Prac*., Vol.8, No.1, pp. 3-12.

Yang, Y.Y. and Linkens, D.A., 1994, Adaptive neural network based approach for the control of continuously stirred tank reactor, *IEE Proc.* Part.D, Vol.141, No.5, pp. 341-349.

Yingwei, L., Sundararajan, N. and Saratchandran, P., 1997, Identification of time-varying nonlinear systems using minimal radial basis function neural networks, *IEE Proc.* Part.D, Vol.144, No.2, pp. 202-208.

Yu, D.L., Gomm, J.B. and Williams, D., 1997, A recursive orthogonal least squares algorithm for training RBF networks. *Neural Processing Letters*, Vol.5, pp 167-176.

Yu, D.L., Gomm, J.B. and Williams, D., 2000, Neural model input selection for a MIMO chemical process, *Engn. Appl. of Artif. Intell*. Vol.13, No.1, pp. 15-23.

Yu, D.L. and Gomm, J.B., 2001, RBFN model adaptation based on orthogonal decomposition, *Proc. IFAC Workshop: Adapt. Learn. In Cont. Signal Proce*. Italy, 8.2001, pp.101-106.

Yu, D.L., Gomm, J.B. and Williams, D., 1999, On-line predictive control of a chemical process using neural network models, *Proc.14th IFAC Congress*, Beijing, pp. 121-126.